1Б

УДК 004.825

В.И. Булкин
Макеевский экономико-гуманитарный институт, Украина
86157, Украина, Донецкая обл., г. Макеевка, ул. Островского, 16, *bulkin01@mail.ru*

Представление алгебропредикатных структур в виде ассоциативно-логических преобразователей

V.I. Bulkin
Makeevka Economic-Humanitarian Institute
86157, Ukraine, Donetsk region, Ostrovskogo str., 16.

Presentation of Algebra-Predicate Structures as Associative-Logic Converters

В.І. Булкін
Макіївський економіко-гуманітарний інститут, Україна
86157, Україна, Донецька обл., м. Макіївка, вул. Островського, 16.

Представлення алгебропредикатних структур у вигляді асоціативно-логічних перетворювачів

В работе рассмотрены вопросы представления алгебропредикатных структур в виде ассоциативно-логических преобразователей. Предложена структурная схема элемента узнавания предмета, состоящая из шифратора, на вход которого подаются имена предметов из универсума предметов, и ячейки ассоциативной памяти, в которой записан код предмета. Разработана структурная схема узнавания множества значений предметной переменной. Показано, что алгебропредикатные структуры можно рассматривать как ассоциативно-логические преобразователи, моделирующие функции интеллекта, связанные с распознаванием предметов и логической обработкой данных.
Ключевые слова: теория интеллекта, алгебра предикатов, ассоциативная память, алгебропредикатные структуры.

In the article, the problems of representation of algebra-predicate structures in the form of associative-logic converters are considered. The structure scheme for the element of recognition of object is offered. The scheme consists of a coder with the entered to its input names of objects from universe of objects and of a cell of associative memory where the code of object is recorded. The structure scheme for a cognizance of a set of values of an object variable is developed. It is shown that algebra-predicate structures can be considered as the associative-logical converters, which model functions of intelligence concerned to object recognition and logic data processing.
Key Words: theory of intelligence, algebra of predicates, associative memory, algebra-predicate structures.

У роботі розглянуто питання представлення алгебропредикатних структур у вигляді асоціативно-логічних перетворювачів. Запропонована структурна схема елемента впізнавання предмета, що складається з шифратора, на вхід якого подаються імена предметів з універсуму предметів, і комірки асоціативної пам’яті, в якій записаний код предмета. Розроблена структурна схема впізнавання множини значень предметної змінної. Показано, що алгебропредикатні структури можна розглядати як асоціативно-логічні перетворювачі, що моделюють функції інтелекту, пов’язані з розпізнаванням предметів і логічною обробкою даних.
Ключові слова: теорія інтелекту, алгебра предикатів, асоціативна пам'ять, алгебропредикатні структури.

Введение

Теорией интеллекта называют науку о математическом описании детерминированных, дискретных и конечных функций человеческого интеллекта [1]. Описание таких функций в рамках теории интеллекта осуществляется с помощью алгебро-предикатных уравнений. Особенностью этих уравнений является наличие большого числа многозначных предметных переменных, связанных между собой логическими операциями. Решать уравнения теории интеллекта можно двумя способами. Первый из них – это решение уравнений на универсальных ЭВМ последовательного действия с использованием специализированного программного обеспечения – так называемых решателей предикатных уравнений. Второй способ решения предикатных уравнений – это аппаратный метод. Любой формуле алгебры предикатов можно поставить в соответствие переключательную цепь или как ее еще называют – алгебропредикатную структуру (АП-структуру). Если схему переключательной цепи реализовать аппаратно, то решение соответствующего ей уравнения будет заключаться в подаче на ее входы наборов значений предметных переменных и последующем снятии с ее выходов соответствующих значений предиката.

В случае применения этого метода, во-первых, отпадает необходимость в использовании дорогостоящих и медленно работающих компьютеров последовательного действия и, как следствие, исчезает необходимость в разработке сложного программного обеспечения для создания решателя предикатных уравнений. Во-вторых, получаемые устройства представляют собой одновременно и модель, и моделирующее устройство, которое может стать основой создания интеллектуальных систем логического типа в виде мозгоподобных структур, моделирующих функции интеллекта без использования алгоритмических и арифметических методов обработки данных. АП-структуры включают в свой состав не только логические элементы, выполняющие булевы операции, но и элементы узнавания предметов, реализующие базовые предикаты вида $x_j^{a_i}$, на входы которых подаются значения предметных переменных. О том, как конструировать эти ключевые элементы, входящие в состав АП-структур, в литературе по теории интеллекта практически ничего не сказано.

Постановка задачи

В данной работе предлагается строить элементы узнавания предметов на основе блоков ассоциативной памяти, которые позволяют представлять k-значные коды имен предметов в виде двоичных кодов и распознавать эти коды при сравнении с содержимым ячеек ассоциативной памяти. В работе [2] высказывается идея убрать из компьютера «тяжеловесную» арифметику и создать мозгоподобную инфраструктуру ассоциативной логики. Для этого предлагается разработать алгебру ассоциативно-логических вычислений в виде специализированной алгебры векторной логики. Однако следует отметить, что уже разработана и существует универсальная алгебра ассоциативно-логических вычислений. Называют эту алгебру алгеброй конечных предикатов [1]. Абстрактной алгеброй конечных предикатов называется множество M, элементами которого являются 0 и 1, а также элементы a_{ij} ($i = 1, 2, \ldots, k$; $j = 1, 2, \ldots, n$). На множестве M определены двухместные операции сложения и умножения, удовлетворяющие всем аксиомам дистрибутивной решетки с нулем и единицей и следующим тождествам:

$$a_{1j} + a_{2j} + \ldots + a_{kj} = 1 \ (j = 1, 2, \ldots, n);$$

$$a_{i_1 j} a_{i_2 j} = 0, \ i_1 \neq i_2; \ i_1 = 1, 2, \ldots, k; \ i_2 = 1, 2, \ldots, k; \ j = 1, 2, \ldots, n.$$

Эти тождества называют аксиомами абстрактной алгебры конечных предикатов. Для перехода от абстрактной алгебры конечных предикатов к конкретной необходимо множество M интерпретировать как множество всевозможных n-местных k-ичных предикатов, элементы 0 и 1 – как тождественно ложный и тождественно истинный предикаты, элементы a_{ij} – как предикаты $x_j^{a_i}$ [3].

Предикат $x_j^{a_i}$ «узнает» произвольно выбранный из множества M предмет x_j, сравнивая его с предметом a_i. Если $x_j = a_i$, то он отождествляет x_j с предметом a_i, сигнализируя об этом положительным ответом $x_j^{a_i} = a_i^{a_i} = 1$. Если же $x_j = b_i$ ($b_i \neq a_i$), то предикат $x_j^{a_i}$ отличает предмет b_i от предмета a_i, сигнализируя об этом отрицательным ответом $x_j^{a_i} = b_i^{a_i} = 0$. Формально это записывают следующим образом [4]:

$$x_j^{a_i} = \begin{cases} 1, \textit{ если } x_j = a_i, \\ 0, \textit{ если } x_j \neq a_i. \end{cases}$$

Описанный выше алгоритм узнавания предмета, а также формальная запись базисного предиката узнавания предмета представляют собой не что иное, как алгоритм и формальную запись работы ассоциативной памяти. В одной из первых работ, посвященных обзору проблем, связанных с ассоциативной памятью, дается следующее определение: «Под ассоциативной памятью обычно понимается некоторый набор, или совокупность, элементов, обладающих способностью хранить информацию. Доступ к этим элементам осуществляется одновременно и параллельно в соответствии с содержанием хранящихся в них данных, а не путем задания адреса или расположения элемента» [5]. Ассоциативную память нередко называют «память с адресацией по содержанию» (ПАС). В работе [6] приводится пример элементарного варианта ПАС, который называют памятью-каталогом (рис. 1).

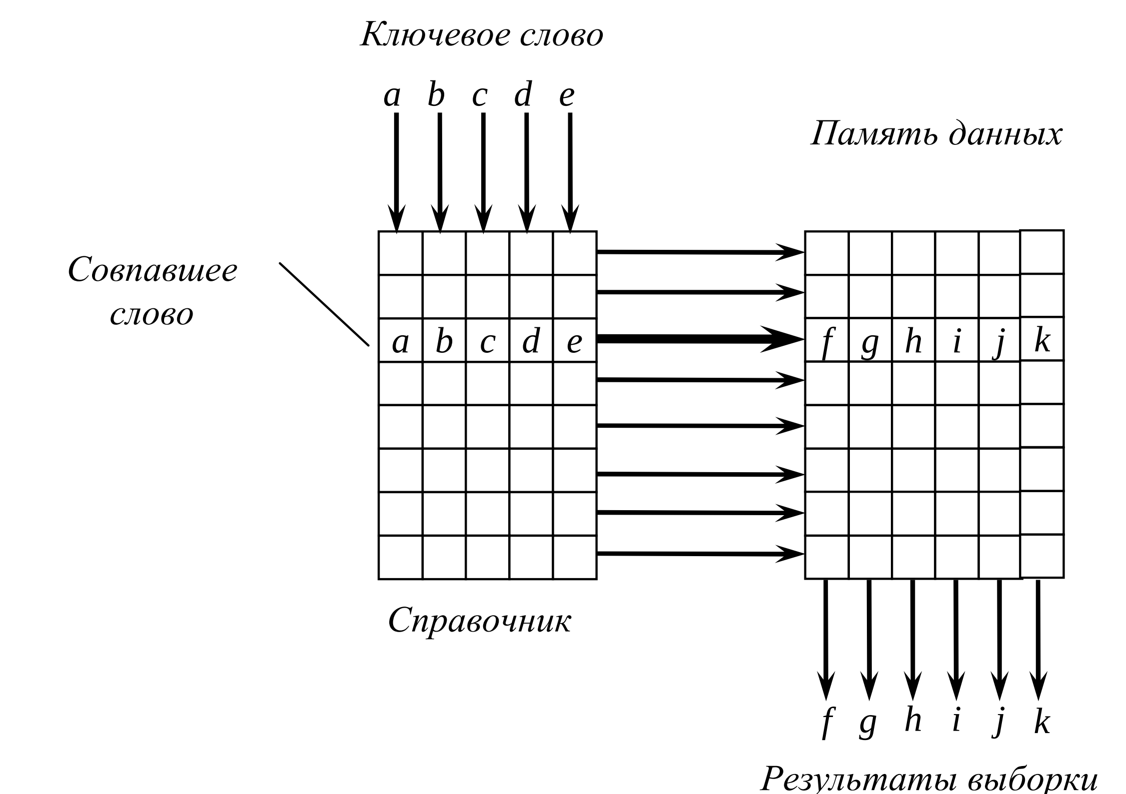


Рисунок 1 – Память-каталог

Данная память стоит из двух частей: справочника и памяти данных. Такая конфигурация допускает сравнение ключевого слова одновременно со всеми словами, записанными в справочнике. Полагают, что все эти слова различны, поэтому совпасть с ключевым словом может только одно из них. Каждая ячейка справочника представляет собой регистр, в котором хранится одно слово. Его дополняют комбинационными логическими схемами, предназначенными для сравнения (компарации) содержимого регистра с ключевым словом, поступающим на все регистры одновременно. В каждой ячейке есть выходная линия, которая возбуждается при совпадении ключевого слова с ее содержимым. При этом возбуждается не более чем одна линия (на рис. 1 – утолщенная линия). Память данных представляет собой обычную память произвольного доступа с линейной выборкой. Выходные шины справочника выполняют функцию адресных шин памяти данных, поэтому дешифратор адреса здесь не нужен. Время доступа к такой памяти очень мало и может составлять не более 50 нс.

Существует два основных метода организации ассоциативной памяти. Первый из них основан на разграничении памяти по содержанию и осуществляется программным путем. Программный метод реализации ПАС называют методом хеширования.

Второй способ связан с применением специальных аппаратных средств, предназначенных для хранения и поиска данных. Остановимся более подробно на аппаратном способе построения ассоциативной памяти. Рассмотрим организацию выборки из памяти по содержанию из справочника (рис. 1). Предполагается, что в каждой ячейке справочника хранится слово в двоичном коде. На рис. 2 изображены логическая схема одной ячейки, входящей в состав справочника. Набор одноразрядных параллельных шин служит для передачи поискового аргумента на входы соответствующих разрядов всех регистров справочника одновременно. Логическая схема ячейки ассоциативной памяти включает в свой состав триггеры для хранения одного бита информации, компараторы, реализующие логическую функцию эквивалентности (по одному на каждый разряд) и многовходовую схему «И», общую для всех разрядов ячейки [6].

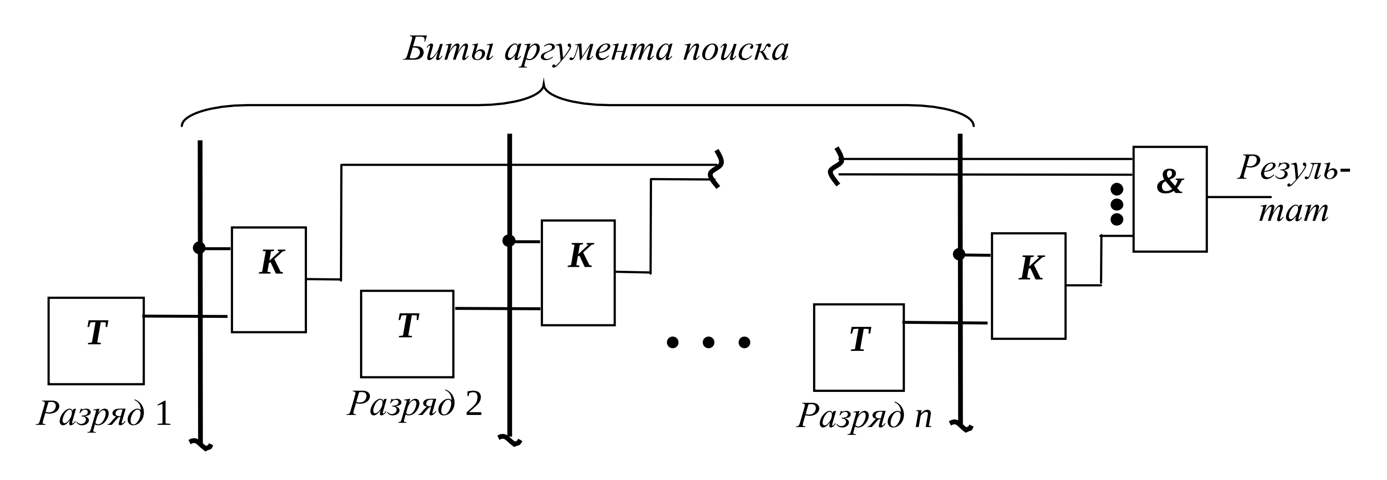


Рисунок 2 – Логическая схема одной ячейки

T – триггер для хранения одного бита информации; *K* – компаратор; & – схема

Основной операцией в запоминающих устройствах с адресацией по совпадению является операция побитового сравнения. Если обозначить логические переменные символами x и y, то для сравнения их значений следует использовать булеву функцию эквивалентности, которая принимает значение 1 (истина), если переменные x и y совпадают, и 0 (ложь) в противном случае. Значения функции эквивалентности представлены в ее таблице истинности (табл. 1).

Таблица 1

x	y	$x \sim y$
0	0	1
0	1	0
1	0	0
1	1	1

В аналитической форме функция эквивалентности может быть записана следующим образом:

$$x \sim y = (x \wedge y) \vee (\bar{x} \wedge \bar{y}). \quad (1)$$

Логическая схема, реализующая соотношение (1), представлена на рис. 3.

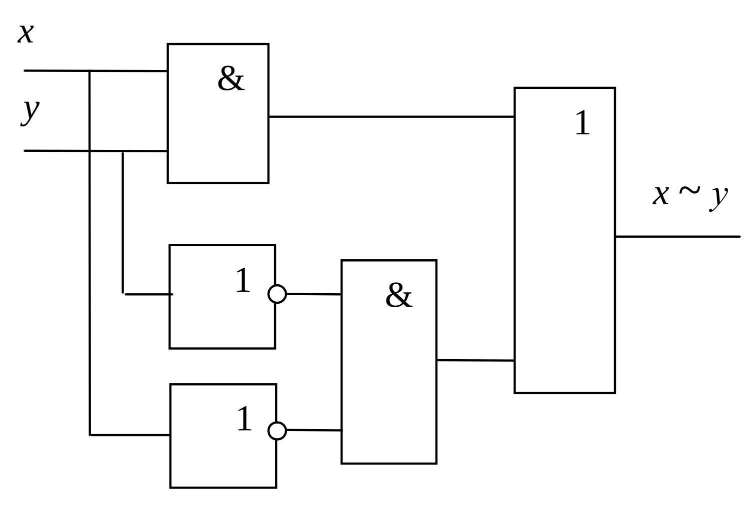


Рисунок 3 – Логическая схема, реализующая функцию эквивалентности

Схема, изображенная на рис. 3, имеет три ступени и включает в свой состав пять логических элементов: две схемы совпадения, два инвертора и схему разделения. С целью упрощения этой схемы преобразуем соотношение (1) следующим образом:

$$x \sim y = (x \wedge y) \vee (\bar{x} \wedge \bar{y}) = (x \wedge y) \vee \overline{\overline{(\bar{x} \wedge \bar{y})}} = (x \wedge y) \vee \overline{(x \vee y)}. \quad (2)$$

Логическая схема, реализующая соотношение (2), имеет две ступени и включает в свой состав всего три логических элемента: одну схему совпадения, одну схему разделения с отрицанием и одну схему разделения без отрицания (рис. 4).

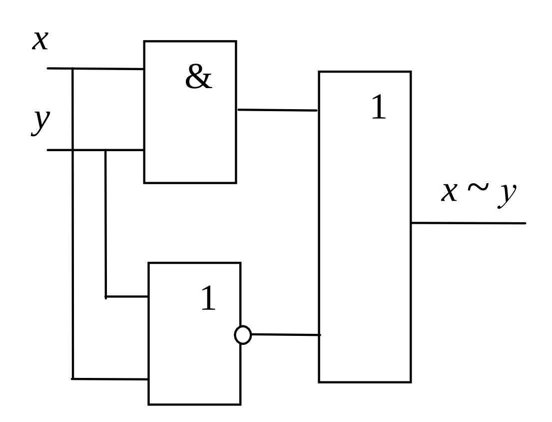


Рисунок 4 – Логическая схема одноразрядного компаратора

Данная схема (рис. 4) представляет собой компаратор, выполняющий операцию сравнения в каждом разряде ячейки ассоциативного запоминающего устройства (АЗУ).

Решение поставленной задачи

Как мы уже упоминали, по формулам алгебры предикатов строятся переключательные цепи (АП-структуры), реализующие соответствующие им предикаты так же, как по формулам булевой алгебры строят комбинационные схемы, реализующие булевы функции [1]. Переключательные цепи строят из элементов четырех типов: элементов узнавания предметов, элементов совпадения, элементов разделения и инверторов. Схемы элементов совпадения, разделения и инверторов были разработаны конструкторами вычислительной техники еще на заре ее зарождения. Элемент узнавания буквы (предмета) реализует предикат узнавания предмета $p = x^a$ (рис. 5).

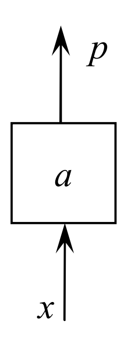


Рисунок 5 – Элемент узнавания буквы (предмета)

В работе [1] отмечается, что элемент узнавания буквы (предмета) *a* должен быть сконструирован так, чтобы при подаче на его вход предмета *a*, он вырабатывал на своем выходе сигнал 1, как бы «узнавая» этот предмет. Исходя из этого требования, самым простым вариантом реализации элемента узнавания предмета может стать ячейка ассоциативной памяти, схема которой представлена на рис. 2. И, действительно, если на вход ячейки ассоциативного запоминающего устройства (АЗУ) подать зашифрованное имя предмета *a*, то на выходе этой ячейки появится сигнал 1 в случае, если в этой ячейке записан код имени данного предмета. Если на вход ячейки подается код другого предмета, например, предмета *b* ($b \neq a$), то на выходе ячейки будет выработан сигнал 0. Исходя из этого, элемент узнавания буквы (предмета) должен включать в свой состав шифратор, кодирующий имена предметов принадлежащих универсуму предметов, и ячейку ассоциативной памяти, в которой записан двоичный код имени предмета *a*. Логические схемы ячейки АЗУ распознают код этого предмета (рис. 6).

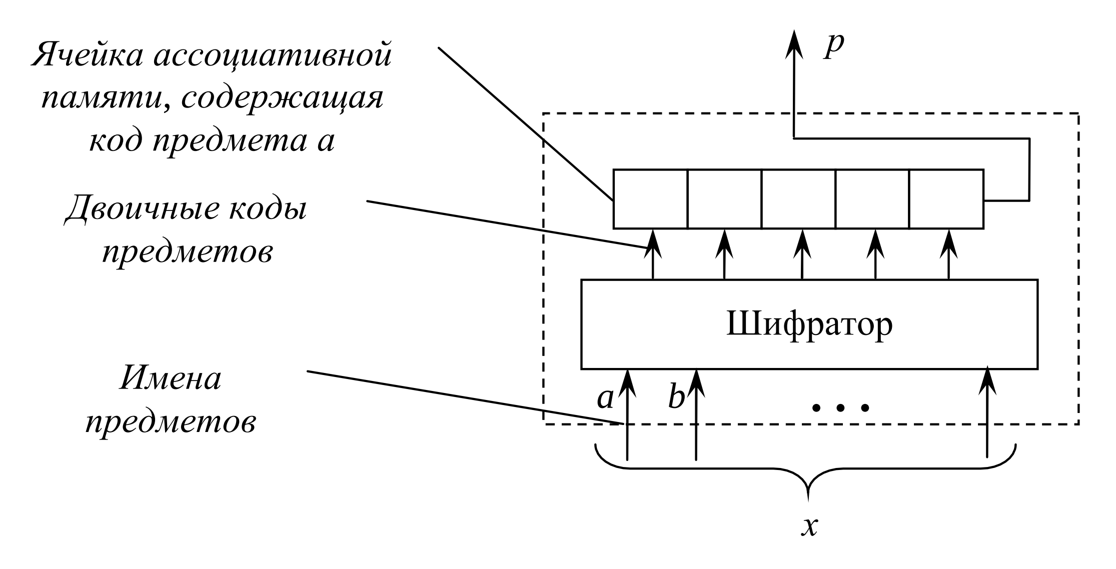


Рисунок 6 – Структура элемента узнавания буквы (предмета) *a*

Схема работает следующим образом. Если на вход шифратора, обозначенный именем предмета a, будет подан единичный сигнал, то на выходе шифратора появится двоичный код этого имени, который после сравнения с содержимым ячейки АЗУ позволит выработать на выходе элемента узнавания буквы (предмета) a сигнал $p = 1$. При подаче сигнала, равного 1, на любой другой вход шифратора, соответствующий именам остальных предметов универсума предметов, на выходе шифратора будет выработан двоичный код имени предмета, который в результате сравнения с содержимым ячейки АЗУ сформирует на выходе элемента узнавания буквы (предмета) a сигнал, равный 0.

Если предметная переменная имеет несколько значений, то схема узнавания множества значений этой переменной будет иметь вид, показанный на рис. 7.

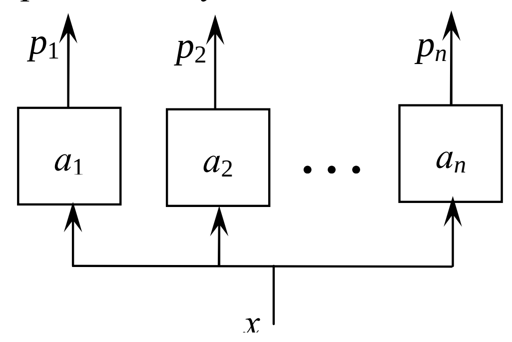


Рисунок 7 – Схема узнавания множества значений предметной

Для реализации схемы узнавания множества значений предметной переменной воспользуемся схемой, в состав которой входит шифратор имен предметов $a_1, a_2, \ldots, a_k$ из универсума предметов, а также набор ячеек ассоциативной памяти, в каждой из которых записан двоичный код имени одного из предметов $a_1, a_2, \ldots, a_n$ (рис. 8).

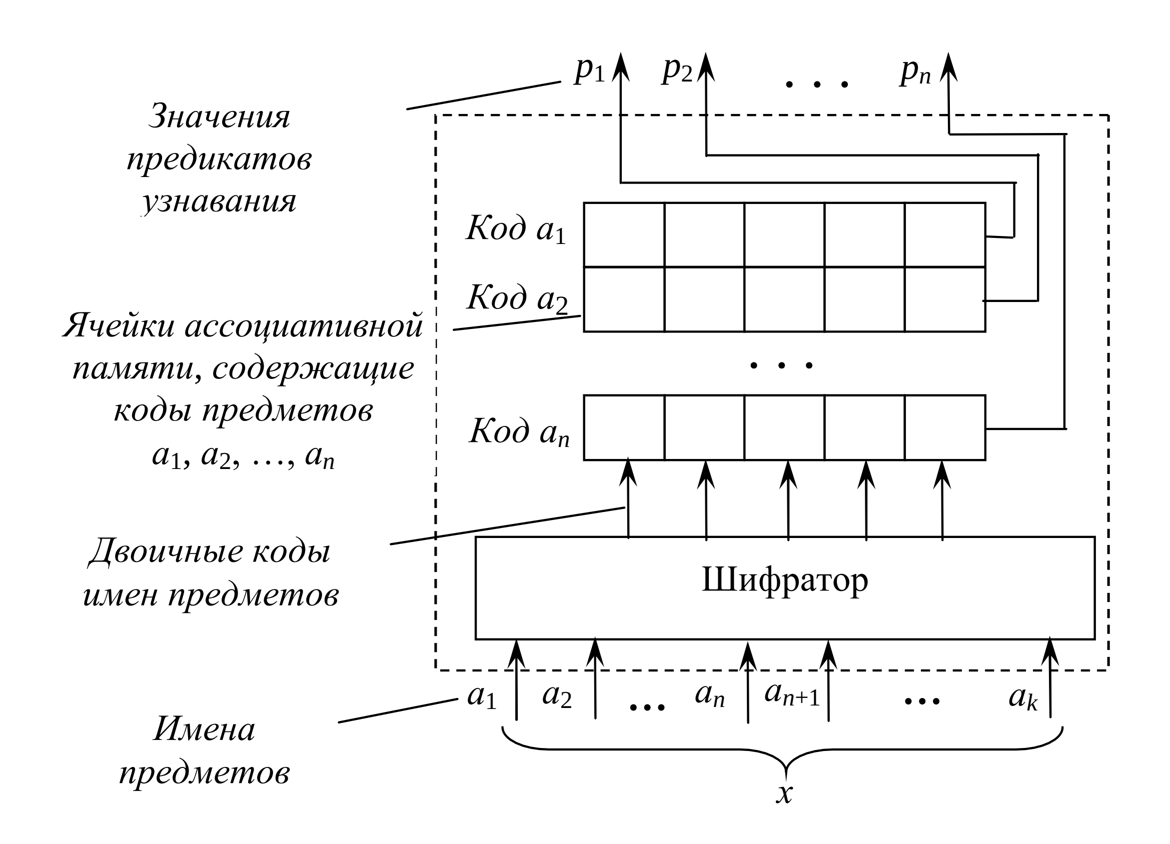


Рисунок 8 – Структурная схема узнавания множества значений предметной переменной

Данная схема работает аналогично предыдущей (рис. 6). Если на один из входов шифратора, обозначенный именем a_i одного из предметов $a_1, a_2, \ldots, a_n$, будет подан единичный сигнал, то на выходе шифратора появится двоичный код этого имени, который после сравнения с содержимым соответствующей ячейки АЗУ сформирует на выходе схемы узнавания значений предметной переменной сигнал $p_i = 1$ $(i = \overline{1, n})$. При подаче сигнала, равного 1, на любой другой вход шифратора, соответствующий именам $a_{n+1}, a_{n+2}, \ldots, a_k$ универсума предметов, на выходе шифратора будет выработан двоичный код имени предмета, который в результате сравнения с содержимым ячейки АЗУ сформирует на выходах схемы узнавания значений предметной переменной сигналы $p_1 = p_2 = \ldots = p_n = 0$.

Любой конечный предикат $y = P(x_1, x_2, \ldots, x_n)$ может быть представлен схемой, включающей в свой состав два блока – блок узнавания предметов и блок логических операций (рис. 9)

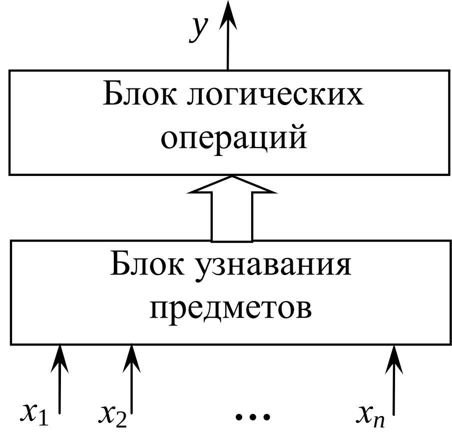
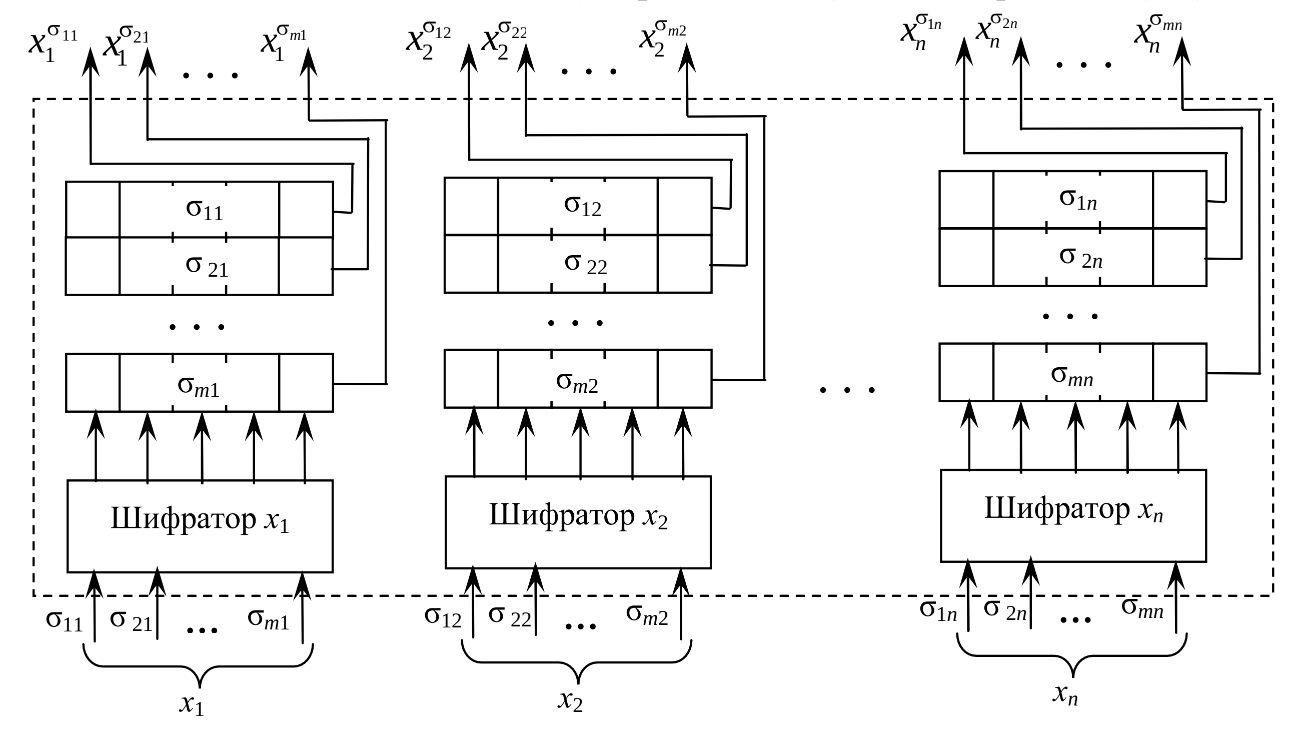


Рисунок 9 – Структурная схема конечного предиката

В свою очередь блок узнавания предметов можно представить в виде набора, состоящего из n блоков ассоциативной памяти, каждый из которых должен хранить двоичные коды имен предметов универсума предметов $U = \{a_1, a_2, \ldots, a_k\}$. Блок логических операций представляет собой комбинационную схему, которая включает в свой состав логические элементы совпадения (И), разделения (ИЛИ) и отрицания (НЕ).



Рисунк 10 – Схема блока узнавания предметов СДНФ предиката

В работе [1] отмечается, что любой конечный предикат может быть представлен в виде совершенной дизъюнктивной нормальной формы (СДНФ):

$$P(x_1,\ x_2,...,\ x_n) = x_1^{\sigma_{11}} x_2^{\sigma_{12}} \ldots x_n^{\sigma_{1n}} \vee x_1^{\sigma_{21}} x_2^{\sigma_{22}} \ldots x_n^{\sigma_{2n}} \vee \ldots \vee x_1^{\sigma_{m1}} x_2^{\sigma_{m2}} \ldots x_n^{\sigma_{mn}} \quad (3)$$

Исходя из формулы (3) схема блока узнаваний предметов СДНФ предиката должна включать в свой состав *n* блоков ассоциативной памяти, на входы шифраторов которых подаются значения предметных переменных $x_1, x_2, \ldots, x_n$ (рис. 10).

Рассмотрим следующий пример. Пусть некоторое предметное пространство задано универсумом предметных переменных $V = \{x_1, x_2\}$ и универсумом предметов $U = \{a, b, c\}$. Пусть предикат $P(x_1, x_2)$ представлен своей СДНФ в следующем виде: $P(x_1, x_2) = x_1^a x_2^b \vee x_1^a x_2^c \vee x_1^b x_2^a$. Алгебропредикатная структура, построенная по этой формуле, показана на рис. 11.

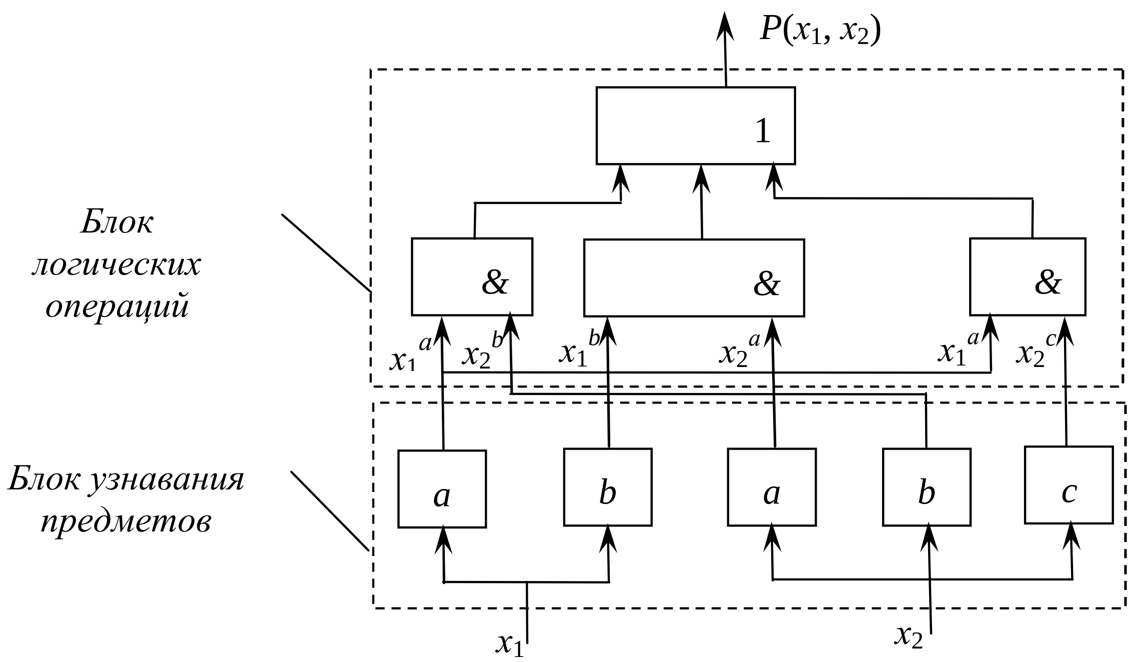


Рисунок 11 – Алгебропредикатная структура СДНФ предиката

Построим теперь блок узнавания предметов (рис. 11) с помощью соответствующих блоков ассоциативной памяти. Блок логических операций оставим в неизменном виде. Тогда алгебропредикатная структура СДНФ предиката $P(x_1, x_2)$ с использованием блоков АЗУ будет выглядеть так, как показано на рис. 12.

Поскольку мощность универсума предметов $|U| = 3$, то для кодирования имен предметов *a*, *b*, *c* достаточно использовать шифраторы значений предметных переменных x_1 и x_2, имеющие по два выхода и, соответственно, двухразрядные ячейки АЗУ, хранящие двоичные коды имен предметов. Двоичные коды имен предметов могут быть следующими: *a* – 01, *b* – 10, *c* – 11. Если, например, на вход *a* шифратора x_1 поступает единичный сигнал, то на выходе шифратора будет сформирован код 01, который при сравнении с содержимым ячеек АЗУ выработает на выходе сигналы $x_1^a =$ $= 1$, $x_1^b = 0$. Аналогично при подаче на вход *b* шифратора x_1 единичного сигнала на выходах ячеек АЗУ будут сформированы сигналы $x_1^a = 0$, $x_1^b = 1$. Если же на вход *c* шифратора x_1 будет подан сигнал, равный 1, то на его выходе появится код 11, который при сравнении с содержимым ячеек АЗУ сформирует на их выходах сигналы $x_1^a =$ $= 0$, $x_1^b = 0$. Подобным образом осуществляется распознавание значений *a*, *b*, *c* предметной переменной x_2.

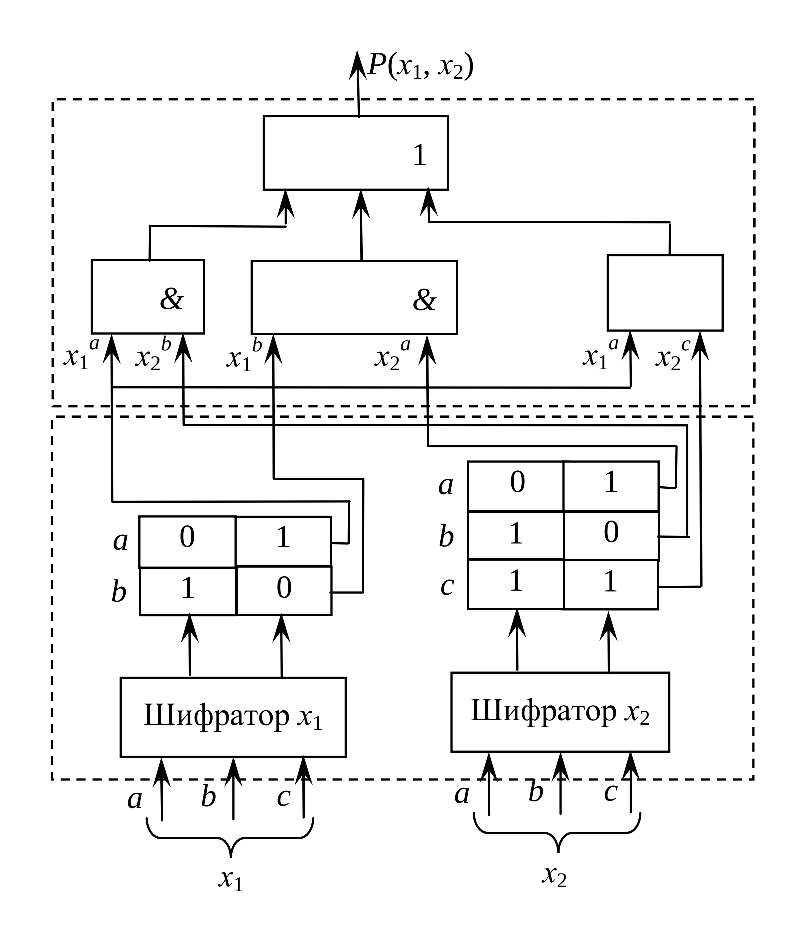


Рисунок 12 – Алгебропредикатная структура СДНФ предиката Р(x_1, x_2) с использованием блоков АЗУ

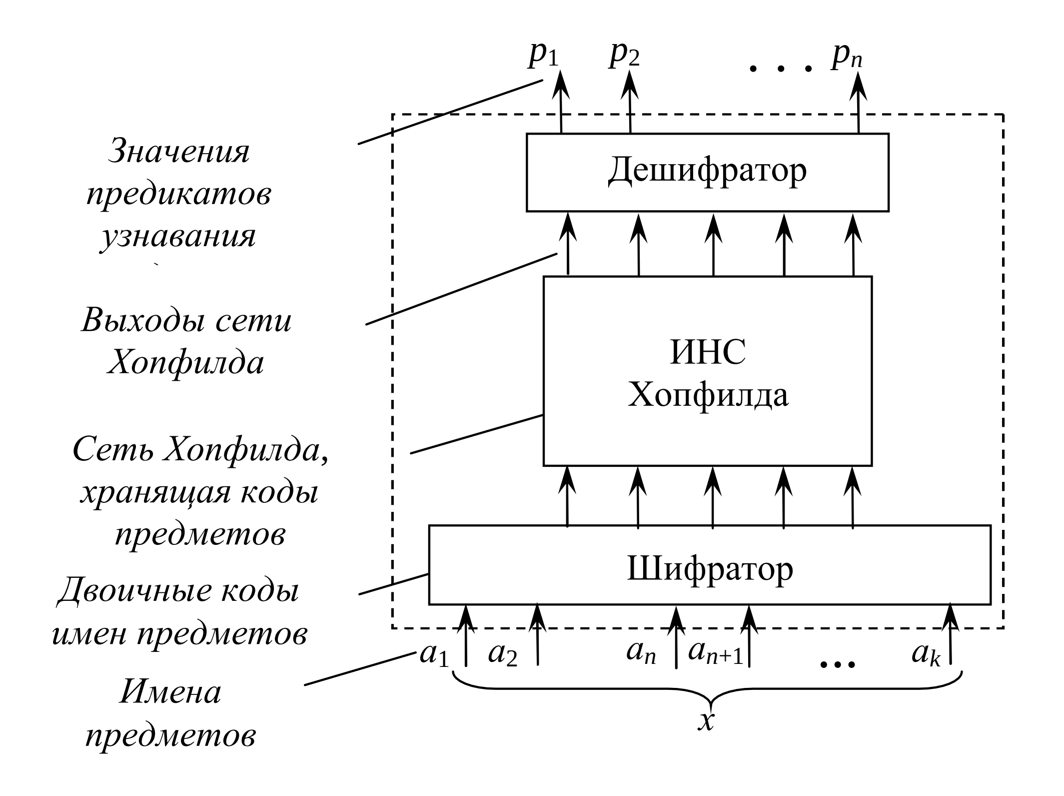


Рисунок 13 – Структурная схема узнавания множества значений предметной переменной на основе сети Хопфилда

Кроме того, ассоциативная память может быть построена на основе искусственных нейронных сетей Хопфилда и Хэмминга [7]. В отличие от сетей с прямым распространением сигналов (Feedforward Networks) сети Хопфилда являются *рекуррентными* сетями или сетями с *обратными связями* (Feedback Networks). Сеть Хопфилда имеет всего один слой, состоящий из n нейронов. Сеть имеет n входов и n выходов. Выход каждого нейрона подается на входы всех остальных нейронов. Обратные связи на свой собственный вход отсутствуют. ИНС Хопфилда может быть обучена для хранения образов в виде двоичных кодов. При подаче на вход сети Хопфилда одного из образов, которые она хранит, на ее выходе формируется двоичный код этого образа. Для определения номера образа, поданного на вход сети, к ее выходу необходимо подключить дешифратор. Таким образом, сеть Хопфилда с дешифратором на выходе может быть использована в качестве ассоциативной памяти для построения элементов узнавания предметов АП-структур. Структурная схема узнавания множества значений предметной переменной на основе сети Хопфилда имеет вид, показанный на рис. 13.

Сеть Хэмминга также может быть использована для организации ассоциативной памяти при создании элементов узнавания предметов АП-структур. При подаче на вход нейронной сети Хэмминга двоичного кода одного из образов, хранимых ИНС, на выходе, соответствующем номеру этого образа, формируется сигнал, равный 1. На остальных выходах формируется сигнал, равный 0. Следовательно, нейронная сеть Хэмминга идеально подходит для построения элементов узнавания предметов на основе АЗУ. Сеть Хэмминга имеет ряд преимуществ по сравнению с сетью Хопфилда. В отличие от сети Хопфилда, емкость сети Хемминга не зависит от размерности входного сигнала, она в точности равна количеству нейронов M в выходном слое [8]. Структурная схема узнавания множества значений предметной переменной на основе сети Хэмминга показана на рис. 14.

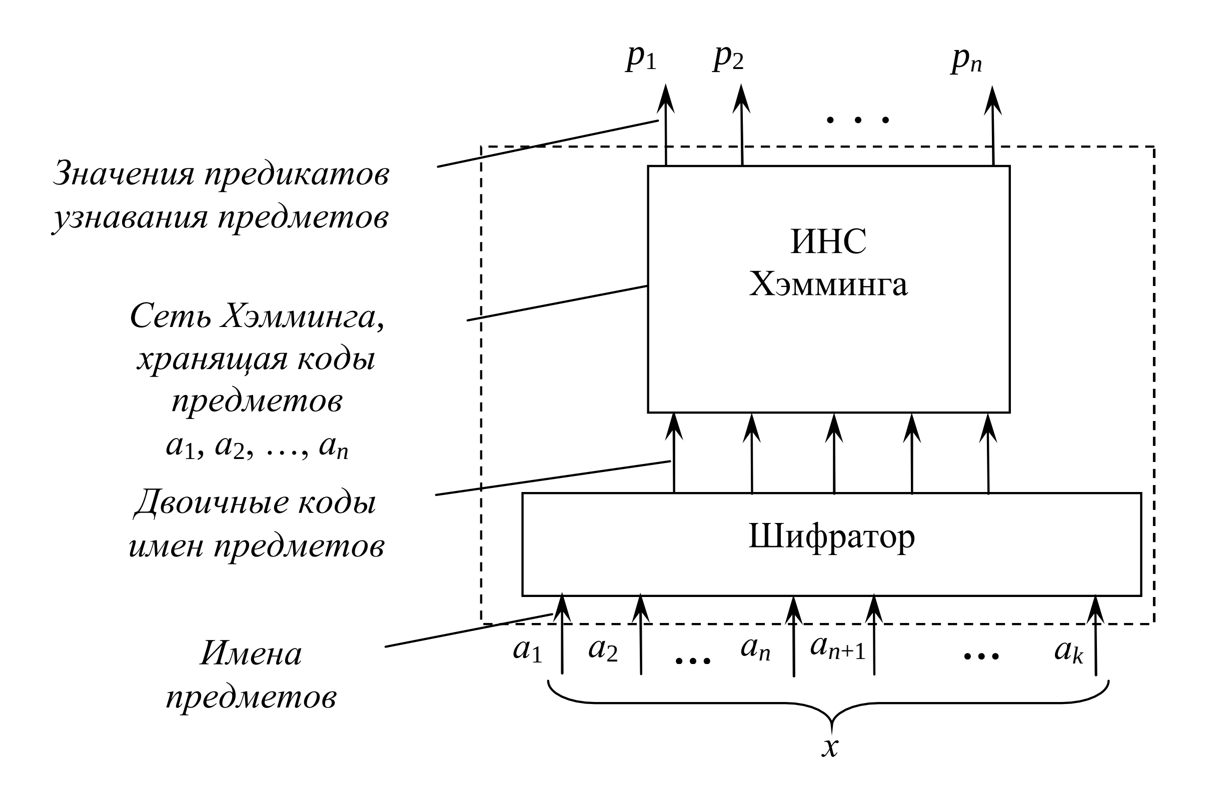


Рисунок 14 – Структурная схема узнавания множества значений предметной переменной на основе сети Хэмминга

К входам и той и другой сети необходимо подключать шифраторы, преобразующие имена предметов в их двоичные коды.

Заключение

В данной работе рассмотрены вопросы представления алгебропредикатных структур в виде ассоциативно-логических преобразователей. Предложена структурная схема элемента узнавания предмета *a*, состоящая из шифратора, на вход которого подаются имена предметов из универсума предметов, и ячейки ассоциативной памяти, в которой записан код предмета *a*. Разработана структурная схема узнавания множества значений предметной переменной. Показано, что любой *n*-местный *k*-ичный предикат может быть представлен схемой, состоящей из блока узнавания предметов и блока логических операций. Построена схема блока узнавания предметов СДНФ предиката. Рассмотрен пример создания АП-структуры СДНФ предиката с использованием блоков ассоциативной памяти. Для построения ассоциативной памяти в составе АП-структур предлагается использовать искусственные нейронные сети Хопфилда и Хэмминга. Таким образом, алгебропредикатные структуры можно рассматривать как ассоциативно-логические преобразователи, моделирующие функции интеллекта, связанные с распознаванием предметов и логической обработкой данных. Можно предположить, что и человеческий интеллект подобным образом использует ассоциативную память для узнавания предметов и логической обработки полученных данных.

Литература

1. Шабанов-Кушнаренко Ю.П. Теория интеллекта. Математические средства / Шабанов-Кушнарен ко Ю.П. – Х. : Вища шк. Изд-во при Харьковском ун-те, 1984. – 144 с.
2. Инфраструктура мозгоподобных вычислительных процессов / [Бондаренко М.Ф., Гузь О.А., Хаханов В.И., Шабанов-Кушнаренко Ю.П.]. – Харьков : ХНУРЭ, 2010.– 160 с.
3. Бондаренко М.Ф. Об абстрактном определении алгебры конечных предикатов / М.Ф. Бондаренко, Ю.П. Шабанов-Кушнаренко // Проблемы бионики. – 1987. – Вып. 39. – С. 3-12.
4. Бондаренко М.Ф. Об алгебре предикатов / М.Ф. Бондаренко, Ю.П. Шабанов-Кушнаренко // Бионика интеллекта. – 2004. – № 1. – С. 15-26.
5. Hanlon A. C. IEEE Trans. – 1966. – EC-15. – P. 509-521.
6. Кохонен Т. Ассоциативные запоминающие устройства : пер. с англ. / Кохонен Т. – М. : Мир, 1982. – 384 с., ил.
7. Руденко О.Г. Искусственные нейронные сети : [учебное пособие] / О.Г. Руденко, Е.В. Бодянский. – Харьков : ООО «Компания Смит», 2005. – 408 с.
8. Электронный ресурс. – Режим доступа : http://i-intellect.ru/articles-of-neural-networks/network-of-hamming.html

Literatura

1. Shabanov-Kushnarenko Y.P. Teorija intellekta. Matematicheskie sredstva. Kh.: Vyscha shk. Izd-vo pri Kharkovskom un-te. 1984. 144 s.
2. Bondarenko M.F. Infrastruktura mozgopodobnyh vychislitel’nyh processov. Kharkov: KNURE. 2010. 160 S.
3. Bondarenko M.F. Problemy bioniki. Kh.: “Vyscha shkola”. 1987. Vyp. 39. S. 3-12.
4. Bondarenko M.F. Bionika intellect. Kh.: Izd-vo KNURE. 2004. № 1. S. 15-26.
5. Hanlon A. C., IEEE Trans., EC-15, 509-521 (1966).
6. Kohonen T. Associativnye zapominajushhie ustrojstva. M.: Mir. 1982. 384 s.
7. Rudenko O.G., Bodianskij E.V. Iskusstvennye nejronnye seti: uchebnoe posobie. Kharkov: OOO “Kompania Smit”. 408 s.
8. http://i-intellect.ru/articles-of-neural-networks/network-of-hamming.html

REZUME

V.I. Bulkin

Presentation of Algebra-Predicate Structures as Associative-Logic Converters

In the article, the problems of representation of algebra-predicate structures in the form of associative-logic converters are considered. The structure scheme for the element of recognition of object is offered. The scheme consists of a coder with the entered to its input names of objects from universe of objects and of a cell of associative memory where the code of object is recorded. The structure scheme for a cognizance of a set of values of an object variable is developed. It is shown that any *n*-place predicate can be presented by the scheme consisting of a block of cognizance of objects and a block of logic operations. The scheme of the block of a cognizance of objects of PDNF predicate is constructed. The example of creation of AP-structure for PDNF of a predicate with use of blocks of associative memory is considered. For creation of associative memory as a part of AP-structures it is offered to use Hopfield’s and Hamming's artificial neural networks. Thus, algebra-predicate structures can be considered as the associative-logical converters, which model functions of intelligence concerned to object recognition and logic data processing.

Статья поступила в редакцию 31.05.2012.